pfSense - Bug #10254

pf error "too many elements" when attempting to load large tables

02/11/2020 02:35 PM - Jim Pingle

Status:	Resolved	**Start date:**	02/11/2020
Priority:	Urgent	**Due date:**	
Assignee:	Renato Botelho	**% Done:**	100%
Category:	Operating System	**Estimated time:**	0.00 hour
Target version:	2.4.5		
Affected Version:	2.4.5	**Affected Architecture:**	All

Description

On at least pfSense-base-2.4.5.r.20200210.0912 and later, pf fails to load large tables no matter what the limits are in pf:

```
: pfctl -f /tmp/rules.debug
/tmp/rules.debug:23: cannot define table bogonsv6: too many elements.
Consider increasing net.pf.request_maxcount.
pfctl: Syntax error in config file: pf rules not loaded
```

However, that OID is not present on 2.4.5:

```
: sysctl net.pf
net.pf.source_nodes_hashsize: 8192
net.pf.states_hashsize: 32768
: sysctl -a | grep request_maxcount
0
:
```

There is plenty of room in the table hard limit:

```
: wc -l /etc/bogonsv6
  108611 /etc/bogonsv6
: pfctl -sm | grep table
table-entries hard limit  2000000
```

Similar to #9356 on 2.5.0, but in that case we set a higher default for that OID. That does not appear to be possible on 2.4.5.

Tried on amd64 and SG-3100, same result on both.

Associated revisions

Revision da569f45 - 02/14/2020 12:52 PM - Renato Botelho

Ticket #10254: Set net.pf.request_maxcount on upgrade

Add pre-install script to pfSense-rc to set default value to
net.pf.request_maxcount before reboot

Revision 9bdf3477 - 02/14/2020 12:53 PM - Renato Botelho

Ticket #10254: Set net.pf.request_maxcount on upgrade

Add pre-install script to pfSense-rc to set default value to
net.pf.request_maxcount before reboot

Revision 3b6ad495 - 02/20/2020 10:25 AM - Renato Botelho

Fix #10254: Default value is minimumtableentries_bogonsv6 from globals.inc

Revision ce164bb8 - 02/20/2020 10:25 AM - Renato Botelho

Fix #10254: Default value is minimumtableentries_bogonsv6 from globals.inc

History

#1 - 02/11/2020 02:36 PM - Jim Pingle

- Target version set to 2.4.5

#2 - 02/11/2020 02:36 PM - Jim Pingle

- Priority changed from Normal to Urgent

#3 - 02/11/2020 02:39 PM - Jim Pingle

The easiest way to reproduce the problem is to enable blocking of Bogons on any interface with IPv6 configured.

#4 - 02/11/2020 03:06 PM - Jim Pingle

Looking in the FreeBSD source, it appears that the code which produces the error (r343520) is present on the branch used for 2.4.5 and in FreeBSD stable/11 but the code which handles the sysctl OID and backend is not (https://reviews.freebsd.org/D15018, 205176451d5ad5f9fc9540f650e9d7efd1f728f5, rS332404).

It would probably be safer to revert the code producing the error than to pull in the larger change. Without that, the error appears to be cosmetic, and if we pull in the other change then we also have to worry about resolving #9356 for 2.4.5.

#5 - 02/12/2020 11:10 AM - Jim Pingle

Current snapshots have that change reverted but are still not behaving properly. Even though there appears to be sufficient room in the table space, pf is yielding a memory allocation error:

```
: wc -l /etc/bogonsv6
  108654 /etc/bogonsv6
: pfctl -sm
states        hard limit   202000
src-nodes     hard limit   202000
frags         hard limit     5000
table-entries hard limit   400000
: pfctl -f /tmp/rules.debug
/tmp/rules.debug:20: cannot define table bogonsv6: Cannot allocate memory
pfctl: Syntax error in config file: pf rules not loaded
```

Similar behavior on amd64 and ARM, but on amd64 it prints an error once and then works the next time, while ARM never works. Rebooting amd64 in this state yields one instance of this allocation error recorded but the table is loaded after boot. Rebooting ARM in this state yields two instances of this error at boot but the ruleset still fails to reload even manually.

Similar configurations work on 2.5.0 with both amd64 and ARM. Tables are loaded, no errors.

#6 - 02/12/2020 11:28 AM - Jim Pingle

Looks to be failing around 65k, which was the default limit on net.pf.request_maxcount

```
: pfctl -T flush -t bogonsv6
: head -n 65535 /etc/bogonsv6-stock > /etc/bogonsv6
: pfctl -f /tmp/rules.debug
/tmp/rules.debug:20: cannot define table bogonsv6: Cannot allocate memory
pfctl: Syntax error in config file: pf rules not loaded
: pfctl -T flush -t bogonsv6
: head -n 65534 /etc/bogonsv6-stock > /etc/bogonsv6
: pfctl -f /tmp/rules.debug
: pfctl -T show -t bogonsv6 | wc -l
   65533
```

#7 - 02/12/2020 11:42 AM - Jim Pingle

https://github.com/pfsense/FreeBSD-src/commit/8f7d14d3049de4fb6f82c7e97153c4372674a1e7 might need to be reverted, or we should just sync up with what 12.x has for net.pf.request_maxcount which is probably safer at this point.

#8 - 02/14/2020 08:02 AM - Jim Pingle

Current snapshots have the code which allows us to set the request limit via net.pf.request_maxcount. However, it isn't being set until late in the upgrade process so the first full post-upgrade boot doesn't have a high enough value to allow bogonsv6 to load without errors.

amd64 first post-upgrade boot:

```
: grep net.pf.request_maxcount /boot/loader.conf
net.pf.request_maxcount="2000000"
: sysctl net.pf.request_maxcount
net.pf.request_maxcount: 500000
```

After reboot:

```
: grep net.pf.request_maxcount /boot/loader.conf
net.pf.request_maxcount="2000000"
: sysctl net.pf.request_maxcount
net.pf.request_maxcount: 2000000
```

SG-3100 first post-upgrade boot (loading bogonsv6 failed):

```
: grep net.pf.request_maxcount /boot/loader.conf
net.pf.request_maxcount="400000"
: sysctl net.pf.request_maxcount
net.pf.request_maxcount: 65535
```

SG-3100 after one more reboot (loading bogonsv6 worked):

```
: grep net.pf.request_maxcount /boot/loader.conf
net.pf.request_maxcount="400000"
: sysctl net.pf.request_maxcount
net.pf.request_maxcount: 400000
```

Looks like we might need to copy or move the code which sets that value to a place that runs earlier, like when the kernel itself gets upgraded or just after the upgrade starts before it reboots the first time.

#9 - 02/20/2020 10:35 AM - Renato Botelho

- Status changed from New to Feedback

- % Done changed from 0 to 100

Applied in changeset 3b6ad495670ca387127dbf72cefb46d909be4fa9.

#10 - 02/20/2020 03:42 PM - Jim Pingle

- Status changed from Feedback to In Progress

- Assignee set to Renato Botelho

Something is still not quite right with this value post-upgrade. The first boot after any firmware upgrade (like one snapshot to the next) fails to use the correct value. Later reboots are fine.

```
: grep request_max /boot/loader.conf
net.pf.request_maxcount="400000"
: sysctl net.pf.request_maxcount
net.pf.request_maxcount: 65535
: pfctl -f /tmp/rules.debug
/tmp/rules.debug:20: cannot define table bogonsv6: too many elements.
Consider increasing net.pf.request_maxcount.
pfctl: Syntax error in config file: pf rules not loaded
```

That was set in loader.conf before the upgrade, so somehow it is either being ignored or cleared/reset during the upgrade.

This is on ARM (SG-3100) but I see a similar issue on amd64 as well.

#11 - 02/21/2020 07:42 AM - Renato Botelho

- Status changed from In Progress to Feedback

pfSense-upgrade 0.74 (on 2.5.0 and 2.4.5) and 0.63 on 2.4.4 will fix it

#12 - 02/23/2020 12:20 PM - Jim Pingle

- Status changed from Feedback to In Progress

There is still a problem here we're investigating

#13 - 02/27/2020 01:47 PM - Renato Botelho

- Status changed from In Progress to Feedback

- pfSense-upgrade was copying loader.conf to a tmp file before upgrade kernel/rc and copying it back to place after that due to a bug that happened in the past where kernel package was installing a static version of loader.conf
- Reverted that and even after that we noted pieces missing from loader.conf during the upgrade
- Noted SG-3100 kernel package still contains a static version of loader.conf. It means we need the pfSense-upgrade hack back, so I revert the reverted commit and added it back
- Removed loader.conf from non-amd64 archs kernel packages
- Reworked pfSense-upgrade to update rc package before backup loader.conf

We are going to make more tests when new snapshots are available. pfSense-upgrade 0.76 must be used

#14 - 03/03/2020 07:51 AM - Jim Pingle

Systems where this problem was due to loader.conf issues appear to be OK on current snapshots. I've upgraded a system which saw the problem on every upgrade in the past and it is OK now.

There is another situation which appears to be similar but isn't the same issue. That has been moved to #10310

#15 - 03/03/2020 07:51 AM - Jim Pingle

- Status changed from Feedback to Resolved

#16 - 09/02/2020 02:04 AM - Dmitry Fill

Just upgraded today to 2.5.0.a.20200901.2100, hitting exact same issue. Seems like regression.

Every reboot have to run:

```
sysctl -w net.pf.request_maxcount=262144
net.pf.request_maxcount: 65535 -> 262144
```

even /boot/loader.conf has value from UI

```
cat /boot/loader.conf
kern.cam.boot_delay=10000
kern.ipc.nmbclusters="1000000"
kern.ipc.nmbjumbop="524288"
kern.ipc.nmbjumbo9="524288"
autoboot_delay="3"
net.pf.request_maxcount="500000"
hw.hn.vf_transparent="0"
hw.hn.use_if_start="1"
```

#17 - 09/02/2020 08:21 AM - Jim Pingle

This issue is quite old and resolved in a previous version. I created a new issue for the regression after confirming it:
https://redmine.pfsense.org/issues/10861